



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Segmenting Subtitles for Correcting ASR Segmentation Errors

Citation for published version:

Wan, D, Kedzie, C, Ladhak, F, Turcan, E, Galušáková, P, Zotkina, E, Jiang, Z, Bell, P & McKeown, K 2021, Segmenting Subtitles for Correcting ASR Segmentation Errors. in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, pp. 2842-2854, 16th conference of the European Chapter of the Association for Computational Linguistics , Virtual Conference, 19/04/21. <<https://www.aclweb.org/anthology/2021.eacl-main.248>>

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Publisher's PDF, also known as Version of record

Published In:

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Segmenting Subtitles for Correcting ASR Segmentation Errors

David Wan¹, Chris Kedzie¹, Faisal Ladhak¹, Elsbeth Turcan¹,
Petra Galuščáková², Elena Zotkina², Zhengping Jiang¹,
Peter Bell³, Kathleen McKeown¹

¹ Columbia University, ² University of Maryland, ³ University of Edinburgh
{dw2735, zj2265}@columbia.edu, {petra, elena}@umiacs.umd.edu,
{kedzie, faisal, eturcan, kathy}@cs.columbia.edu,
peter.bell@ed.ac.uk

Abstract

Typical ASR systems segment the input audio into *utterances* using purely acoustic information, which may not resemble the sentence-like units that are expected by conventional machine translation (MT) systems for Spoken Language Translation. In this work, we propose a model for correcting the acoustic segmentation of ASR models for low-resource languages to improve performance on downstream tasks. We propose the use of subtitles as a proxy dataset for correcting ASR acoustic segmentation, creating synthetic acoustic utterances by modeling common error modes. We train a neural tagging model for correcting ASR acoustic segmentation and show that it improves downstream performance on MT and audio-document cross-language information retrieval (CLIR).

1 Introduction

Typical ASR systems segment the input audio into *utterances* using purely acoustic information, i.e., pauses in speaking or other dips in the audio signal, which may not resemble the sentence-like units that are expected by conventional MT systems for spoken language translation (SLT) (Cho et al., 2017). Longer utterances may span multiple sentences, while shorter utterances may be sentence fragments containing only a few words (see Figure 1 for examples). Both can be problematic for downstream MT systems. In this work, we propose a model for correcting the acoustic segmentation of an ASR model to improve performance on downstream tasks, focusing on the challenges inherent to SLT pipelines for low-resource languages.

While prior work has trained intermediate components to segment ASR output into sentence-like units (Matusov et al., 2007; Rao et al., 2007), these have primarily focused on highly resourced language pairs such as Arabic and Chinese. When the source language is low-resource, suitable training data may be very limited for ASR and MT,

and even nonexistent for segmentation. Since typical low-resource language ASR datasets crawled from the web do not have hand-annotated segments, we propose deriving proxy segmentation datasets from film and television subtitles. Subtitles typically contain segment boundary information like sentence-final punctuation, and while they are not exact transcriptions, they are closer to transcribed speech than many other large text corpora.

Our proposed model takes as input a sequence of tokens and segmentation boundaries produced by the acoustic segmentation of the ASR system and returns a corrected segmentation. While subtitles are often similar to speech transcripts, they lack an existing acoustic segmentation for our model to correct. To account for this, we generate synthetic acoustic segmentation by explicitly modeling two common error modes of ASR acoustic segmentation: under- and over-segmentation.

We evaluate the downstream MT performance in a larger SLT pipeline, and show improvements in translation quality when using our segmentation model to correct the acoustic segmentation provided by ASR. We also extrinsically evaluate our improved SLT pipeline as part of a document-level cross-lingual information retrieval (CLIR) task, where we show that improvements in ASR segmentation also lead to improved relevance of search results. We report results for nine translation settings: Bulgarian (BG) to English, Lithuanian (LT) to English, and Farsi (FA) to English, and when using either phrase-based, statistical MT (SMT) or one of two neural MT (NMT) models. We finally perform an ablation study to examine the effects of our synthetic acoustic boundaries and our over- and under-segmentation noise.

This paper makes the following contributions. (i) We propose the use of subtitles as a proxy dataset for correcting ASR acoustic segmentation and (ii) a method for adding synthetic acoustic utterance segmentations to a subtitle dataset, as well as (iii) a

Acoustic Segmentation (Over-segmentation):
ARE YOU OKAY ■ AGENT SCULLY ■ YOU KIND OF SOUNDED A ■ LITTLE SPOOKY ■
Corrected Sentence Segmentation:
ARE YOU OKAY AGENT SCULLY ■ YOU KIND OF SOUNDED A LITTLE SPOOKY ■
Acoustic Segmentation (Under-segmentation):
NO IS HE IN SOME KIND OF TROUBLE ■
Corrected Sentence Segmentation:
NO ■ IS HE IN SOME KIND OF TROUBLE ■

Figure 1: Example acoustic segmentation errors and their corrections. ■ indicates a segment boundary.

simple neural tagging model for correcting ASR acoustic segmentation before use in an MT pipeline. (iv) Finally, we show downstream performance increases on MT and document-level CLIR tasks, especially for more syntactically complex segments.

2 Related Work

Segmentation in SLT has been studied quite extensively in high-resource settings. Early work used kernel-based SVM models to predict sentence boundaries using language model probabilities along with prosodic features such as pause duration (Matusov et al., 2007; Rao et al., 2007) and part-of-speech features derived from a fixed window size (Rangarajan Sridhar et al., 2013). Other work has modeled the problem using hidden markov models (Shriberg et al., 2000; Gotoh and Renals, 2000; Christensen et al., 2001; Kim and Woodland, 2001) and conditional random fields (Liu et al., 2005; Lu and Ng, 2010).

More recent segmentation work uses neural architectures, such as LSTM (Sperber et al., 2018) and Transformer models (Pham et al., 2019). These models benefit from the large training data available for high-resource languages. For example, the TED corpus (Cettolo et al., 2012) for SLT from English to German includes about 340 hours of well-transcribed data. To our knowledge, such datasets do not exist for the languages we are interested in. Wan et al. (2020) develop a segmentation model in our setting using subtitles; however, they do not take into account explicit modeling of segmentation errors and show only minimal and intermittent improvements in downstream tasks.

Recent work has increasingly focused on end-to-end models of SLT in a high-resource setting, since these systems reduce error propagation and latency when compared to cascaded approaches

(Weiss et al., 2017; Cross Vila et al., 2018; Sperber et al., 2019; Gaido et al., 2020; Bahar et al., 2020; Lakumarapu et al., 2020). In spite of these advantages, end-to-end systems have only very recently achieved competitive results due to the limited amount of parallel data for speech translation as compared to the data that is available to train ASR systems and translation systems separately (Gaido et al., 2020; Ansari et al., 2020).

3 Problem Definition

We treat the ASR acoustic segmentation problem as a sequence tagging problem (Stolcke and Shriberg, 1996). Unlike a typical tagging problem, which aims to tag a single input sequence, our input is a pair of aligned sequences of n items, $\mathbf{x} = [x_1, \dots, x_n]$ and $\gamma = [\gamma_1, \dots, \gamma_n]$ where \mathbf{x} and γ are the ASR tokens and acoustic segmentation respectively. The tokens x_i belong to a finite vocabulary \mathcal{V} , while the acoustic segmentation boundary tags are binary, i.e., $\gamma_i \in \{0, 1\}$, where $\gamma_i = 1$ indicates that the ASR acoustic segmentation placed a boundary between tokens x_i and x_{i+1} . The goal is to predict a corrected segment boundary tag sequence $\mathbf{y} = [y_1, \dots, y_n] \in \{0, 1\}^n$ from \mathbf{x} and γ .

We do this by learning a probabilistic mapping from token/segmentation sequences to corrected segmentation $p(\cdot | \mathbf{x}, \gamma; \theta) : \{0, 1\}^n \rightarrow (0, 1)$ where p is a neural tagging model with parameters θ . While γ are produced solely from acoustic cues, p can take advantage of both the acoustic information (via γ) as well as syntactic/semantic cues implicit in \mathbf{x} .

Reference Segmentation:										
YEAH ■ THE HOLIDAY MARKET IS TOO BUSY ■ YES ■										
Synthetic Acoustic Segmentation Generation:										
Model Output:	y	=	1	0	0	0	0	0	1	1
(Under-seg.)	$\tilde{\gamma}$	=	1	—	—	—	—	—	0	1
(Over-seg.)	$\hat{\gamma}$	=	—	0	0	1	0	1	—	—
Model Input:	γ	=	1	0	0	1	0	1	0	1
	x	=	YEAH	THE	HOLIDAY	MARKET	IS	TOO	BUSY	YES
Synthetic Acoustic Segmentation (γ):										
YEAH ■ THE HOLIDAY MARKET ■ IS TOO ■ BUSY YES ■										

Figure 2: Example of synthetic acoustic segmentation (γ) creation. For each training datapoint, we have as model input the tokens x and the corresponding model output sentence boundary labels y . To generate the synthetic acoustic segmentation (γ), we apply under-segmentation ($\tilde{\gamma}$) and over-segmentation ($\hat{\gamma}$) noise to y . Dashes indicate the tokens where the particular noise is not applicable. Bold indicates the changed labels due to the noise. We generate the additional input γ by combining both $\tilde{\gamma}$ and $\hat{\gamma}$.

4 Generating Training Data from Subtitles

One of our primary contributions is a method for converting subtitle data into suitable training data for an ASR segmentation correction model. The subtitle data contains speech-like utterances of dialogue between characters in film and television shows. For the purposes of this paper, we do not use information about speaker identity, only the text and information about segmentation. We obtain the ground truth output label segmentation y by segmenting the subtitle text on sentence final punctuation.¹ We remove the punctuation but keep the implied label sequence to obtain the input token sequence x and ground truth output label segmentation y . However, we do not have acoustic segmentation available for x , y pairs derived from subtitle data, which we will need as additional input if our model is to learn to correct acoustic segmentation provided by an ASR component. We thus create a synthetic acoustic segmentation sequence γ as input by adding two types of noise to y . Specifically, we imitate two common ASR system errors, under-segmentation noise and over-segmentation, so that at test time the model can correct those errors.

Under-segmentation Noise In the ASR model, under-segmentation occurs when pauses between words are brief, and the resulting ASR output is an utterance that could ideally be split into multiple sentence-like segments. We simulate this by adding

under-segmentation noise which converts ground truth segmentation boundaries $y_i = 1$, to $y_i = 0$ with probability $\tilde{\alpha}$ and leaves $y_i = 0$ unchanged.

Over-segmentation Noise Over-segmentation occurs in an ASR model when a speaker takes a longer pause in the middle of what could be interpreted as a contiguous sentence-like utterance. *Over-segmentation noise* is simulated by inserting random segment boundaries within an utterance. That is, with probability $\hat{\alpha}$ we convert a non-boundary tag $y_i = 0$ to $y_i = 1$, while leaving all $y_i = 1$ unchanged.

Synthetic Segmentation Input Generation We can then sample a synthetic acoustic segmentation sequence γ from the following distribution,

$$\gamma_i \sim \begin{cases} \text{Bernoulli}(\tilde{\alpha}) & \text{if } y_i = 1 \\ \text{Bernoulli}(\hat{\alpha}) & \text{otherwise} \end{cases}$$

for $i \in \{1, \dots, n\}$. This can be thought of as dropout applied to the correct label sequence y . See Figure 2 for an example. Our proposed segmentation correction model will learn to denoise the input segmentation sequence γ and produce the corrected sequence y .

5 Model

We employ a Long Short-Term Memory (LSTM)-based model architecture for this task (Hochreiter and Schmidhuber, 1997). Given an input sequence of ASR tokens $x = [x_1, \dots, x_n]$ along with corresponding ASR segmentation sequence

¹Set of sentence final punctuation: $\{ () : - ! ? . \}$.

$\gamma = [\gamma_1, \dots, \gamma_n]$, we first get an embedding representation $\mathbf{e}_i \in \mathbb{R}^{316}$ for each token as follows:

$$\mathbf{e}_i = G(x_i) \oplus F(\gamma_i)$$

where $G \in \mathbb{R}^{|\mathcal{V}| \times 300}$ and $F \in \mathbb{R}^{2 \times 16}$ are embedding lookup tables, and \oplus is the concatenation operator. We initialized G with FastText embeddings pre-trained on Common Crawl data (Mikolov et al., 2018). F is randomly initialized.

We pass the embedding sequence through a two-layer bi-directional LSTM, with 512 hidden units each, to get the contextual representation $\mathbf{h}_i \in \mathbb{R}^{1024}$ for each token as follows:

$$\mathbf{h}_i = \overrightarrow{\text{LSTM}}(\mathbf{e}_i) \oplus \overleftarrow{\text{LSTM}}(\mathbf{e}_i)$$

where $\overrightarrow{\text{LSTM}}$ and $\overleftarrow{\text{LSTM}}$ are the forward direction and backward direction LSTMs respectively.

Each output state \mathbf{h}_i is then passed through a linear projection layer with a logistic sigmoid to compute the probability of a segment boundary $p(y_i = 1 | \mathbf{h}_i; \theta)$. The log-likelihood of a corrected segmentation boundary sequence is $\log p(\mathbf{y} | \mathbf{x}, \gamma; \theta) = \sum_{i=1}^n \log p(y_i | \mathbf{h}_i; \theta)$. We fit the parameters, θ , by approximately minimizing the negative log-likelihood on the training set \mathcal{D} , $\mathcal{L}(\theta) = -\frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}, \gamma, \mathbf{y}) \in \mathcal{D}} \log p(\mathbf{y} | \gamma, \mathbf{x}; \theta)$, using mini-batch stochastic gradient descent.

6 Datasets

6.1 Subtitles Dataset

We obtain monolingual subtitle data from the OpenSubtitles 2018 corpus (Lison and Tiedemann, 2016). OpenSubtitles contains monolingual subtitles for 62 languages drawn from movies and television. The number of subtitle documents varies considerably from language to language. LT has only 1,976 documents, while BG and FA have 107,923 and 12,185 respectively. We randomly down-sample from the larger collection to 2,000 documents to ensure our segmentation correction models are all trained with similar amounts of data.

Treating the subtitles for a complete television episode or movie as the source of a single training instance $(\mathbf{x}, \gamma, \mathbf{y})$ introduces some complications because they are usually quite long relative to typical SLT system input. To better match our evaluation conditions, we arbitrarily split each document into M instances, where the length l in tokens for each instance m is sampled from $L \sim U(1, 100)$, i.e. uniformly from 1 to 100 tokens. This range

was determined to approximate the length distribution of our evaluation datasets.

See Table 1 for statistics on the number of training instances created as well as the average number of sentence segments per instance. Note that even though the number of subtitle documents is close to equal, the documents can vary considerably in length. As result, the BG dataset has more than twice the training instances of FA or LT. In some cases, an instance may contain only a few words that do not constitute a sentence, and such instances would have no segment boundaries; this helps prevent the model from learning pathological solutions such as always inserting a segment boundary at the end of the sequence.

Since we do not evaluate the segmentation directly on OpenSubtitles, we split the available data into training and development partitions, with 90% of the instances in the training set.

6.2 Speech Retrieval Dataset

For extrinsic evaluation of ASR segments, we use the speech retrieval dataset from the MATERIAL² program. The goal of MATERIAL is to develop systems that can retrieve text and speech documents in low-resource languages that are relevant to a given query in English. To bootstrap speech retrieval systems in low-resource languages, MATERIAL collects BG, FA, and LT speech training data for ASR systems, as well as additional separate collections of BG, FA, and LT speech documents along with their relevance judgments for a set of English language queries. Since the retrieval of speech documents requires a cascade of ASR, MT, and CLIR systems, the MATERIAL data allows us to measure the impact of ASR segmentation on both the translation quality, as well as the downstream retrieval system. The data partitions in MATERIAL are numerous and to avoid confusion, we briefly describe them here.

The *BUILD* partition contains a small amount of ASR training and development data for BG, FA, and LT, i.e. audio files paired with reference transcripts. We use the BUILD data for fine-tuning our subtitle trained model. We apply the same synthetic acoustic segmentation generation procedure to this collection as we do to the subtitle data when using it for fine-tuning. See Table 1 for dataset statistics.

The *Test (Small)* partition contains audio docu-

²www.iarpa.gov/index.php/research-programs/material

Lang.	OpenSubtitles			BUILD Train			BUILD Valid		
	D	I	S	D	I	S	D	I	S
BG	2000	459,301	3.95	352	7,723	3.22	108	2,472	3.14
FA	2000	120,039	3.88	302	6,707	3.52	120	2,679	3.53
LT	1977	165,751	12.99	484	11,782	3.40	112	2,893	3.31

Table 1: Segmentation model training dataset statistics. We report number of documents (D), number of instances (I) in each dataset and average number of segments per instance (S).

Lang.	Test (Small)		Test (Large)	
	Q	D	Q	D
BG	300	634	–	–
FA	221	528	–	–
LT	300	496	1,000	3,297

Table 2: Number of queries (Q) and documents (D) in the speech retrieval test collections.

ments and a set of English language queries and relevance judgements for those queries. At test time, we use the acoustic segmentation provided by the ASR system as the input γ instead of generating acoustic label sequences. Additionally, roughly half of the audio documents in this collection include ground-truth transcriptions and translations to English, which allows us to evaluate MT.

The *Test (Large)* partition is similar to the *Test (Small)* partition, but much bigger in size. There are no transcripts or translations, so it can be used only to evaluate CLIR. The *Test (Large)* partition is available only for LT.

We use the translated portion of *Test (Small)* as a test set for MT and both *Test (Small)* and *Test (Large)* as extrinsic test sets for CLIR. The statistics of the MATERIAL partitions can be found in Table 2.³

The speech retrieval datasets come from three domains: news broadcast, topical broadcast such as podcasts, and conversational speech from multiple low-resource languages. Some speech documents have two speakers, with each speaker on a separate channel, i.e., completely isolated from the other speaker. When performing segmentation we treat each channel independently, creating a separate (re-segmented) ASR output for each channel. To create the document transcript for MT, we merge the two

³The official MATERIAL collections are named ANALYSIS+DEV and EVAL, but we refer to them as *Test (Small)* and *Test (Large)* to avoid confusion.

output sequences by sorting the token segments based on their wall-clock start time.

7 Experiments

7.1 Segmentation Model Training

For all datasets, we tokenize all data with Moses (Koehn et al., 2007). To improve performance on out of vocabulary words, we use Byte-Pair-Encoding (Sennrich et al., 2016) with 32,000 merge operations to create subwords for each language.

We then train the segmentation model on the subtitle dataset. When creating γ sequences on the subtitles data, we set under- and over-segmentation noise to $\tilde{\alpha} = 0.25$ and $\hat{\alpha} = 0.25$ respectively.⁴ We use the Adam optimizer (Kingma and Ba, 2015) with learning rate of 0.001. We use early stopping on the validation loss of the OpenSubtitles validation set to select the best stopping epoch for the segmentation model.

We further fine-tune this model on the BUILD partition to expose the model to some in-domain training data. The data is similarly prepared as OpenSubtitles. We use early stopping on the development loss of this partition.

7.2 ASR-Segmentation-MT-CLIR Pipeline

We evaluate our segmentation correction model in the context of a CLIR pipeline for retrieving audio documents in BG, FA, or LT that are relevant to English queries. We refer to the three languages BG, FA, and LT as *source languages*. This pipeline uses ASR to convert source language audio documents to source language text transcripts, and MT to translate the source language transcripts into English transcripts. Then a monolingual English IR component is used to return source language documents that are relevant to the issued English queries. We insert our segmentation correction model into

⁴Values for $\tilde{\alpha}$ and $\hat{\alpha}$ were determined by grid-search over $\{0.25, 0.5, 0.75\}$ that minimized loss on the BUILD Valid data.

this pipeline between the ASR and MT components, i.e. (i) ASR to (ii) Segmentation Correction, to (iii) MT to (iv) IR. For clarity we reiterate, the segmentation model takes as input a source language transcript and returns the source language transcript with corrected segmentation.

To implement the ASR, MT, and IR components, we use implementations developed by MATERIAL program participants (Oard et al., 2019).

7.2.1 ASR System

We use the ASR systems developed jointly by the University of Cambridge and the University of Edinburgh (Ragni and Gales, 2018; Carman-tini et al., 2019). The ASR system uses a neural network based acoustic model, trained in a semi-supervised manner on web-scraped audio data, to overcome the small amount of training data in the BUILD data. Separate models are trained for narrow-band audio (i.e., conversational speech) and wide-band audio (i.e. news and topical broadcast).

7.2.2 Segmentation Correction

At test time, given a speech document, the ASR system produces a series of acoustically derived utterances, i.e. $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}$, from this input. In our setting, the corresponding acoustic label sequence $\gamma^{(i)}$ for each utterance would be zero everywhere except the final position, i.e. $\gamma^{(i)} = [0, 0, \dots, 0, 1]$. If we were to process each utterance, $(\mathbf{x}^{(i)}, \gamma^{(i)})$, individually, the model may not have enough context to correct under-segmentation at the ends of the utterance. For example, when correcting the final token position, which by definition will precede a long audio pause, the model will only see the left-hand side of the context. To avoid this, we run our segmentation correction model on consecutive pairs of ASR output utterances, i.e. $(\mathbf{x}^{(i)} \oplus \mathbf{x}^{(i+1)}, \gamma^{(i)} \oplus \gamma^{(i+1)})$. Under this formulation each ASR output utterance is corrected twice (except for the first and last utterances which are only corrected once), therefore we have two predictions $\hat{y}_j^{(i,L)}$ and $\hat{y}_j^{(i,R)}$ for the j -th segment boundary. We resolve these with the logical-OR operation to obtain the final segmentation correction, i.e. $\hat{y}_j^{(i)} = \hat{y}_j^{(i,L)} \vee \hat{y}_j^{(i,R)}$.

Based on the segmentation corrections produced by our model, we re-segment the ASR output tokens and hand the resulting segments off to the MT component where they are individually translated.

7.2.3 MT Systems

We evaluate with three different MT systems. We use the neural MT model developed by the University of Edinburgh (EDI-NMT) and the neural and phrase-based statistical MT systems from the University of Maryland (UMD-NMT and UMD-SMT, respectively). The EDI-NMT and UMD-NMT systems are Transformer-based models (Vaswani et al., 2017) trained using the Marian Toolkit (Junczys-Dowmunt et al., 2018) and Sockeye (Hieber et al., 2018), respectively. UMD-NMT trains a single model for both directions of a language pair (Niu et al., 2018), while EDI-NMT has a separate model for each direction. UMD-SMT is trained using the Moses SMT Toolkit (Koehn et al., 2003), where the weights were optimized using MERT (Och, 2003).

7.2.4 IR System

For the IR system, we use the bag-of-words language model implemented in Indri (Strohman et al., 2005). Documents and queries are both tokenized and normalized on the character level to avoid potential mismatch in the vocabulary. The queries are relatively short, typically consisting of only a few words, and they define two types of relevancy – the conceptual queries require the relevant documents to be topically relevant to the query, while the simple queries require the relevant document to contain the translation of the query. However, no specific processing is used for these two relevance types in our experiments.

7.3 MT Evaluation

Our first extrinsic evaluation measures the BLEU (Papineni et al., 2002) score of the MT output on the Test (Small) sets after running our segmentation correction model, where we have ground truth reference English translations. We refer to our model trained only on the BUILD data as *Seg*, and our subtitle-trained model as *Seg + Sub*. As our baseline, we compare the same pipeline using the segmentation produced by the acoustic model of the ASR system, denoted *Acous*.

Since each segmentation model produces segments with different boundaries, we are unable to use BLEU directly to compare to the reference sentences. Therefore, we concatenate all segments of a document and treat them as one segment, which we refer to as “document-level” BLEU score. We use SacreBLEU⁵ (Post, 2018) with the lowercase

⁵<https://github.com/mjpost/sacrebleu>

option due to the different casing for the reference English translation and MT output.

We also provide BLEU scores for the MT output using the reference transcriptions (Ref) to show the maximum score the system can achieve when there is no ASR or segmentation error. This represents the theoretical upper bound for our pipeline with a perfect ASR system.

Segmentation errors (i.e., the acoustic model incorrectly segmented an utterance) and word errors (i.e., the ASR system produces an incorrect word) can both affect the downstream MT performance. To isolate the segmentation errors from word errors, we align the ASR output tokens to the reference transcriptions by timecode in order to obtain a reference system that has no segmentation errors, but does have transcription errors. This represents a more realistic ceiling for our model because while we can correct segmentation, we cannot correct word errors. We refer to this system in the results section as Align.

7.4 Document-Level CLIR Evaluation

Our second extrinsic evaluation is done on the MATERIAL CLIR task. We are given English queries and asked to retrieve audio documents in either BG, FA, or LT. In our setup, we only search over the English translations of the segmented transcripts produced by our pipeline, i.e., we do not translate the English query into the other languages or search the audio signal directly. We evaluate the performance of CLIR using the Maximum Query Weighted Value (MQWV) from the ground-truth query-relevance judgements for documents in the Test (Small & Large) collections. MQWV, which is a variant of the official MATERIAL program metric called Actual Query Weighted Value (NIST, 2017, AQWV), is a recall-oriented rank metric that measures how well we order the retrieval collection with respect to query relevance.

AQWV is calculated as the average of $1 - (P_m + \beta * P_{fa})$ for each query, where P_m is the probability of misses, P_{fa} is the probability of false alarms, and β is a hyperparameter. The maximum possible value is 1 and the minimum value is given by $-\beta$. In our experiments β it is set to 40. AQWV thus not only depends on the ranking of the documents but also on β .

Additionally, AQWV is sensitive to the threshold used by the IR system to determine document relevance. To avoid the tuning of thresholds, we

Lang.	Model	EDI NMT	UMD NMT	UMD SMT
BG	Acous.	20.48	20.39	21.24
	Seg	22.38*	23.35*	21.23
	Seg + Sub	24.73*	25.92*	21.23
	Align	24.98	27.81	21.29
	Ref	43.75	35.40	29.50
FA	Acous.	5.35	6.26	4.54
	Seg	5.32	6.22	3.28
	Seg + Sub	6.47*	6.83*	4.50
	Align	7.67	7.02	4.59
	Ref	17.08	11.24	7.76
LT	Acous.	15.20	8.38	14.76
	Seg	15.18	8.34	14.76
	Seg + Sub	15.22	8.33	14.76
	Align	15.60	8.71	14.84
	Ref	20.40	11.94	21.30

Table 3: Document-level BLEU scores on ANALYSIS set. * represents statistical significance when compared to Acous. at the 0.05 level. We show the result of translations using the original acoustic segmentation (Acous.), our model trained only on the BUILD dataset (Seg), and our full model (Seg + Sub). For reference, we provide the scores of translation on ASR tokens aligned to the reference transcription segmentation (Align), and the reference transcription (Ref).

report MQWV which is calculated for the optimal threshold; in our experiments this threshold is estimated over the ranks of the documents. Thus, MQWV doesn't depend on the ability to estimate the threshold and only depends on the quality of the document ranking for a given query.

8 Results

8.1 MT

Table 3 shows the results of the MT evaluation. The best non-reference system for each language and MT system is in bold. We compute statistical significance against the acoustic (*Acous.*) segmentation baseline using Welch's T Test (Welch, 1947). Our subtitle-based segmentation model (*Seg + Sub*) consistently improves BLEU scores of NMT models for BG and FA, while not making significant differences in SMT. This echoes prior work (Khayrallah and Koehn, 2018; Rosales Núñez et al., 2019) suggesting SMT models are more robust to noisy inputs than neural models.

Lang.	Model	EDI NMT	UMD NMT	UMD SMT
BG	Acous.	0.173	0.134	0.164
	Seg + Sub	0.177	0.180	0.164
FA	Acous.	0.040	0.039	0.046
	Seg + Sub	0.071	0.042	0.145
LT	Acous.	0.128	0.067	0.157
	Seg + Sub	0.136	0.060	0.172

Table 4: MQWV scores on Test (Small) set.

In 6 out of 9 cases, we see that adding the subtitles data improves over using only the BUILD data. Of the remaining cases, the scores remain similar (i.e., it doesn’t hurt the model). Training on the BUILD data alone improves BG NMT models, but for SMT and the other languages, it either makes no difference or is worse than the acoustic model.

Comparing Seg + Sub with Align in all languages, we see that there is only a small gap between the two. This suggests that our model is nearing the ceiling on what correcting segmentation can do to improve downstream MT. Furthermore, on LT where our model offers only small or no improvement, we see that the original acoustic segmentation is almost performing as well as Align. This suggests that there is relatively little room for improving LT MT by correcting sentence boundaries alone.

8.2 Document-Level CLIR

MQWV on the Test (Small) and Test (Large) partitions are shown in Table 4 and Table 5 respectively. On the Test (Small) partition, we see that our segmentation model improves the CLIR performance over the acoustic segmentation in 7 out of 9 cases. On the Test (Large) partition, we see that our segmentation model improves downstream retrieval performance consistently across all three MT systems. We note that while we measure the downstream retrieval performance separately for each MT system, a real-world CLIR system could perform IR over the union of multiple MT systems, which could yield even further improvements in retrieval performance (Zhang et al., 2020).

8.3 Complexity Analysis

We hypothesize that the effects of improved segmentation should be more pronounced for more complex utterances with more opportunities to mis-

Lang.	Model	EDI NMT	UMD NTM	UMD SMT
LT	Acous.	0.292	0.175	0.328
	Seg + Sub	0.293	0.179	0.399

Table 5: MQWV scores on the Test (Large) set.

ARI Quartile	Model	EDI NMT	UMD NMT
Q1	Acous.	17.78	24.70
	Seg + Sub	17.92	11.62
Q2	Acous.	20.76	22.09
	Seg + Sub	26.89	31.24
Q3	Acous.	22.87	20.38
	Seg + Sub	29.96	33.22
Q4	Acous.	23.41	21.00
	Seg + Sub	29.87	35.24

Table 6: Bulgarian BLEU scores on Test (Small) (transcribed portion) when separated into quartiles by sentence complexity (as measured by ARI).

place boundaries. Therefore, we calculate a measure of sentence complexity, the Automated Readability Index (ARI) (Senter and Smith, 1967), for all documents in Test (Small)⁶ and examine the performance of our *Sub* model on MT. We separate the documents into quartiles based on their calculated ARI, where a higher ARI (and thus a higher quartile) indicates a more complex document, and present the average document-level BLEU score for each quartile in Table 6. In the interest of space, we present results for Bulgarian and for NMT, and defer other languages and SMT to Appendix A. We see that the most dramatic gains in BLEU occur for documents in the third and fourth quartiles, which matches our intuition. In other words, our segmentation model most improves the translation quality of more syntactically complex segments.

9 Ablation Study

We perform an ablation study on two components in our proposed model, (i) the use of acoustic segmentation boundary labels γ as input and (ii) training with a combination of over- and under-segmentation noise. We use the same training and

⁶Only the transcribed portion with reference translations. For each MT system, we compute ARI on the document translation using the reference transcription.

Model	EDI NMT	UMD NMT	UMD SMT
Acous.	20.48	20.39	21.24
Lex.	23.96	24.97	21.58
Lex. + Under	24.27	25.84	21.48
Lex. + Over	24.69	25.88	21.23
Full	24.73	25.92	21.23

Table 7: Document-level BLEU score for the models in ablation studies. We provide the acoustic model (Acous.) and our proposed model (Full).

evaluation process and only modify the affected component. We perform our ablation on the BG MT task, since it had a wider range of improvements than the other languages.

Use of Acoustic Segmentation Boundaries.

We train a segmentation model using only ASR output tokens \mathbf{x} as input without the the ASR segmentation sequence γ . For this model, we modify the embedding representation \mathbf{e}_i so that we do not use F :

$$\mathbf{e}_i = G(x_i)$$

This model, which we refer to as *Lex.*, must exclusively use the lexical information of the ASR token sequence \mathbf{x} to make predictions.

Over-segmentation and Under-segmentation.

The two segmentation problems of the system may have different impact on the MT system. To see their individual effects, we train two models where the synthetic acoustic segmentation boundary sequence γ is created using only under-segmentation or over-segmentation noise. We refer to those models as *Lex. + Under* and *Lex. + Over* respectively.

Results Table 7 shows the effects of the model ablations on MT system BLEU score. On both NMT systems, we see that there is a roughly 1 point improvement on BLEU when including the ASR segmentation boundaries as input. For both NMT models we also find that over-segmentation noise helps slightly more than adding under-segmentation noise, but that these additions are complementary, i.e. the full model does best overall. For SMT, we surprisingly find that model without acoustic segmentation boundary input does best. The overall difference between the acoustic (*Acous.*) baseline and any of the segmentation correction models is small compared to the gains had

on NMT. This again suggests that SMT is more robust to changes in segmentation.

10 Conclusion

We propose an ASR segmentation correction model for improving SLT pipelines. Our model makes use of subtitles data as well as a simple model of acoustic segmentation error to train an improved ASR segmentation model. We demonstrate downstream improvements on MT and CLIR tasks. In future work, we would like to find a better segmentation error model that works well in conjunction with SMT systems in addition to NMT systems.

Acknowledgements

This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract #FA8650-17-C-9117. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes not withstanding any copyright annotation therein.

References

- Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondřej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, Alexander Waibel, and Chanchan Wang. 2020. [FINDINGS OF THE IWSLT 2020 EVALUATION CAMPAIGN](#). In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 1–34, Online. Association for Computational Linguistics.
- Parnia Bahar, Patrick Wilken, Tamer Alkhouli, Andreas Guta, Pavel Golik, Evgeny Matusov, and Christian Herold. 2020. [Start-Before-End and End-to-End: Neural Speech Translation by AppTek and RWTH Aachen University](#). In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 44–54, Online. Association for Computational Linguistics.
- Andrea Carmantini, Peter Bell, and Steve Renals. 2019. [Untranscribed Web Audio for Low Resource Speech Recognition](#). In *Proceedings of Interspeech 2019*, pages 226–230.

- Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. [WIT³: Web Inventory of Transcribed and Translated Talks](#). In *Proceedings of the 16th Annual conference of the European Association for Machine Translation*, pages 261–268, Trento, Italy. European Association for Machine Translation.
- Eunah Cho, Jan Niehues, and Alex Waibel. 2017. [NMT-Based Segmentation and Punctuation Insertion for Real-Time Spoken Language Translation](#). In *Proceedings of Interspeech 2017*, pages 2645–2649.
- Heidi Christensen, Yoshihiko Gotoh, and Steve Renals. 2001. [Punctuation Annotation using Statistical Prosody Models](#). In *Proceedings of ISCA Workshop on Prosody in Speech Recognition and Understanding*, pages 35–40.
- Laura Cross Vila, Carlos Escolano, José A. R. Fonollosa, and Marta R. Costa-Jussà. 2018. [End-to-End Speech Translation with the Transformer](#). In *Proceedings of IberSPEECH 2018*, pages 60–63.
- Marco Gaido, Mattia A. Di Gangi, Matteo Negri, and Marco Turchi. 2020. [End-to-end Speech-Translation with Knowledge Distillation: FBK@IWSLT2020](#). In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 80–88, Online. Association for Computational Linguistics.
- Yoshihiko Gotoh and Steve Renals. 2000. [Sentence Boundary Detection in Broadcast Speech Transcripts](#). In *Proceedings of ISCA Workshop: Automatic Speech Recognition: Challenges for the new Millennium ASR-2000*, pages 228–235.
- Felix Hieber, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2018. [The Sockeye Neural Machine Translation Toolkit at AMTA 2018](#). In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 200–207, Boston, MA. Association for Machine Translation in the Americas.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long Short-Term Memory](#). *Neural Computation*, 9(8):1735–1780.
- Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. [Marian: Fast Neural Machine Translation in C++](#). In *Proceedings of ACL 2018, System Demonstrations*, Melbourne, Australia.
- Huda Khayrallah and Philipp Koehn. 2018. [On the Impact of Various Types of Noise on Neural Machine Translation](#). In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83, Melbourne, Australia. Association for Computational Linguistics.
- Ji-Hwan Kim and Philip C. Woodland. 2001. [The use of Prosody in a Combined System for Punctuation Generation and Speech Recognition](#). In *EUROSPEECH 2001 Scandinavia, 7th European Conference on Speech Communication and Technology, 2nd INTERSPEECH Event, Aalborg, Denmark, September 3-7, 2001*, pages 2757–2760. ISCA.
- Diederik P. Kingma and Jimmy Ba. 2015. [Adam: A method for Stochastic Optimization](#). In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. [Moses: Open Source Toolkit for Statistical Machine Translation](#). In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.
- Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. [Statistical Phrase-Based Translation](#). In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.
- Nikhil Kumar Lakumarapu, Beomseok Lee, Sathish Reddy Indurthi, Hou Jeung Han, Mohd Abbas Zaidi, and Sangha Kim. 2020. [End-to-End Offline Speech Translation System for IWSLT 2020 using Modality Agnostic Meta-Learning](#). In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 73–79, Online. Association for Computational Linguistics.
- Pierre Lison and Jörg Tiedemann. 2016. [OpenSubtitles2016: Extracting Large Parallel Corpora from Movie and TV Subtitles](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Yang Liu, Andreas Stolcke, Elizabeth Shriberg, and Mary Harper. 2005. [Using Conditional Random Fields for Sentence Boundary Detection in Speech](#). In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*, pages 451–458, Ann Arbor, Michigan. Association for Computational Linguistics.
- Wei Lu and Hwee Tou Ng. 2010. [Better Punctuation Prediction with Dynamic Conditional Random Fields](#). In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 177–186, Cambridge, MA. Association for Computational Linguistics.

- Evgeny Matusov, Dustin Hillard, Mathew Magimai-doss, Dilek Hakkani-tur, Mari Ostendorf, and Hermann Ney. 2007. [Improving Speech Translation with Automatic Boundary Prediction](#). In *Proceedings of Interspeech*, pages 2449–2452.
- Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhersch, and Armand Joulin. 2018. [Advances in Pre-Training Distributed Word Representations](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- NIST. 2017. [The Official Original Derivation of AQWV](#).
- Xing Niu, Michael Denkowski, and Marine Carpuat. 2018. [Bi-Directional Neural Machine Translation with Synthetic Parallel Data](#). In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 84–91, Melbourne, Australia. Association for Computational Linguistics.
- Douglas W. Oard, Marine Carpuat, Petra Galuscakova, Joseph Barrow, Suraj Nair, Xing Niu, Han-Chin Shing, Weijia Xu, Elena Zotkina, Kathleen McKeown, Smaranda Muresan, Efsun Selin Kayi, Ramy Eskander, Chris Kedzie, Yan Virin, Dragomir R. Radev, Rui Zhang, Mark J F Gales, Anton Ragni, and Kenneth Heafield. 2019. [Surprise languages: Rapid-Response Cross-Language IR](#). In *Proceedings of the Ninth International Workshop on Evaluating Information Access (EVIA 2019)*, pages 23–27. National Institute of Informatics. The Ninth International Workshop on Evaluating Information Access (EVIA 2019) : a Satellite Workshop of the NTCIR-14 Conference, EVIA 2019 ; Conference date: 10-06-2019 Through 10-06-2019.
- Franz Josef Och. 2003. [Minimum Error Rate Training in Statistical Machine Translation](#). In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [BLEU: a Method for Automatic Evaluation of Machine Translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Ngoc-Quan Pham, Thai-Son Nguyen, Thanh-Le Ha, Juan Hussain, Felix Schneider, Jan Niehues, Sebastian Stüker, and Alexander Waibel. 2019. [The IWSLT 2019 KIT Speech Translation System](#). In *International Workshop on Spoken Language Translation (IWSLT)*.
- Matt Post. 2018. [A Call for Clarity in Reporting BLEU Scores](#). In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Anton Ragni and Mark Gales. 2018. [Automatic Speech Recognition System Development in the “Wild”](#). In *Proceedings of Interspeech 2018*, pages 2217–2221.
- Vivek Kumar Rangarajan Sridhar, John Chen, Srinivas Bangalore, Andrej Ljolje, and Rathinavelu Chengalvarayan. 2013. [Segmentation Strategies for Streaming Speech Translation](#). In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 230–238, Atlanta, Georgia. Association for Computational Linguistics.
- Sharath Rao, Ian Lane, and Tanja Schultz. 2007. [Optimizing Sentence Segmentation for Spoken Language Translation](#). In *Proceedings of Interspeech*.
- José Carlos Rosales Núñez, Djamé Seddah, and Guillaume Wisniewski. 2019. [Comparison between NMT and PBSMT Performance for Translating Noisy User-Generated Content](#). In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 2–14, Turku, Finland. Linköping University Electronic Press.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural Machine Translation of Rare Words with Subword Units](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.
- R.J. Senter and E.A. Smith. 1967. [Automated Readability Index](#). *AMRL-TR. Aerospace Medical Research Laboratories*.
- Elizabeth Shriberg, Andreas Stolcke, Dilek Hakkani-Tür, and Gökhan Tür. 2000. [Prosody-Based Automatic Segmentation of Speech into Sentences and Topics](#). *Speech Communication*, 32(1–2):127–154.
- Matthias Sperber, Graham Neubig, Jan Niehues, and Alex Waibel. 2019. [Attention-Passing Models for Robust and Data-Efficient End-to-End Speech Translation](#). *Transactions of the Association for Computational Linguistics*, 7:313–325.
- Matthias Sperber, Ngoc Quan Pham, Thai Son Nguyen, Jan Niehues, Markus Müller, Thanh-Le Ha, Sebastian Stüker, and Alexander Waibel. 2018. [KIT’s IWSLT 2018 SLT Translation System](#). In *International Workshop on Spoken Language Translation (IWSLT)*.
- Andreas Stolcke and Elizabeth Shriberg. 1996. [Automatic Linguistic Segmentation of Conversational Speech](#). *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP ’96*, 2:1005–1008 vol.2.

- Trevor Strohman, Donald Metzler, Howard Turtle, and W. Croft. 2005. [Indri: A language-model based search engine for complex queries](#). *Information Retrieval*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is All You Need](#). In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.
- David Wan, Zhengping Jiang, Chris Kedzie, Elsbeth Turcan, Peter Bell, and Kathy McKeown. 2020. [Subtitles to Segmentation: Improving Low-Resource Speech-to-Text Translation Pipelines](#). In *Proceedings of the workshop on Cross-Language Search and Summarization of Text and Speech (CLSSTS2020)*, pages 68–73, Marseille, France. European Language Resources Association.
- Ron J. Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. [Sequence-to-sequence Models Can Directly Translate Foreign Speech](#). In *Proceedings of Interspeech 2017*, pages 2625–2629.
- B. L. Welch. 1947. [THE GENERALIZATION OF ‘STUDENT’S’ PROBLEM WHEN SEVERAL DIFFERENT POPULATION VARLANCES ARE INVOLVED](#). *Biometrika*, 34(1-2):28–35.
- Le Zhang, Damianos Karakos, William Hartmann, Manaj Srivastava, Lee Tarlin, David Akodes, Sanjay Krishna Gouda, Numra Bathool, Lingjun Zhao, Zhuolin Jiang, Richard Schwartz, and John Makhoul. 2020. [The 2019 BBN Cross-lingual Information Retrieval System](#). In *Proceedings of the workshop on Cross-Language Search and Summarization of Text and Speech (CLSSTS2020)*, pages 44–51, Marseille, France. European Language Resources Association.

A Full Complexity Analysis

We present the full results of our complexity analysis as described in [subsection 8.3](#). Bulgarian ([Table 8](#)), Lithuanian ([Table 9](#)), and Farsi ([Table 10](#)) results are shown for all three MT models as well as both the acoustic segmentation and our *Seg + Sub* segmentation correction model. The best score for each MT system and quartile is bolded.

ARI Quartile	Model	EDI NMT	UMD NMT	UMD SMT
Q1	Acous.	17.78	24.70	22.26
	Seg + Sub	17.92	11.62	22.58
Q2	Acous.	20.76	22.09	23.35
	Seg + Sub	26.89	31.24	23.43
Q3	Acous.	22.87	20.38	22.45
	Seg + Sub	29.96	33.22	21.68
Q4	Acous.	23.41	21.00	22.97
	Seg + Sub	29.87	35.24	23.25

Table 8: Bulgarian BLEU scores on Test (Small) (transcribed portion) when separated into quartiles by sentence complexity (as measured by ARI).

ARI Quartile	Model	EDI NMT	UMD NMT	UMD SMT
Q1	Acous.	3.46	3.40	3.21
	Seg + Sub	6.05	4.49	3.43
Q2	Acous.	3.55	3.87	3.22
	Seg + Sub	5.06	5.41	3.66
Q3	Acous.	4.71	5.33	5.19
	Seg + Sub	7.44	7.29	5.54
Q4	Acous.	5.88	6.15	3.90
	Seg + Sub	7.81	7.59	4.47

Table 10: Farsi BLEU scores on Test (Small) (transcribed portion) when separated into quartiles by sentence complexity (as measured by ARI).

ARI Quartile	Model	EDI NMT	UMD NMT	UMD SMT
Q1	Acous.	4.24	3.13	5.14
	Seg + Sub	4.32	2.76	5.11
Q2	Acous.	14.07	7.18	13.45
	Seg + Sub	14.01	6.83	13.39
Q3	Acous.	15.95	7.81	14.85
	Seg + Sub	16.39	8.13	14.87
Q4	Acous.	15.41	7.94	15.85
	Seg + Sub	14.60	7.78	15.87

Table 9: Lithuanian BLEU scores on Test (Small) (transcribed portion) when separated into quartiles by sentence complexity (as measured by ARI).